

(12) NACH DEM VERTRAG ÜBER DIE INTERNATIONALE ZUSAMMENARBEIT AUF DEM GEBIET DES PATENTWESENS (PCT) VERÖFFENTLICHTE INTERNATIONALE ANMELDUNG

(19) Weltorganisation für geistiges Eigentum
Internationales Büro(43) Internationales Veröffentlichungsdatum
4. September 2003 (04.09.2003)

PCT

(10) Internationale Veröffentlichungsnummer
WO 03/073301 A1(51) Internationale Patentklassifikation⁷: G06F 13/32

(21) Internationales Aktenzeichen: PCT/DE03/00138

(22) Internationales Anmeldedatum:
20. Januar 2003 (20.01.2003)

(25) Einreichungssprache: Deutsch

(26) Veröffentlichungssprache: Deutsch

(30) Angaben zur Priorität:
102 08 300.2 26. Februar 2002 (26.02.2002) DE

(71) Anmelder (für alle Bestimmungsstaaten mit Ausnahme von US): ROBERT BOSCH GMBH [DE/DE]; Postfach 30 02 20, 70442 Stuttgart (DE).

(72) Erfinder; und

(75) Erfinder/Anmelder (nur für US): BARTHEL, Joerg [DE/DE]; An der Innerste Au 22, 31139 Hildesheim (DE). BEHME, Dirk [DE/DE]; Hachmeisterstrasse 24, 31139 Hildesheim (DE).

(81) Bestimmungsstaaten (national): AT, AU, BR, CA, CH, CZ, ES, FI, GB, IN, JP, KR, LU, MX, PL, RU, SE, SK, US, ZA.

(84) Bestimmungsstaaten (regional): europäisches Patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, SE, SI, SK, TR).

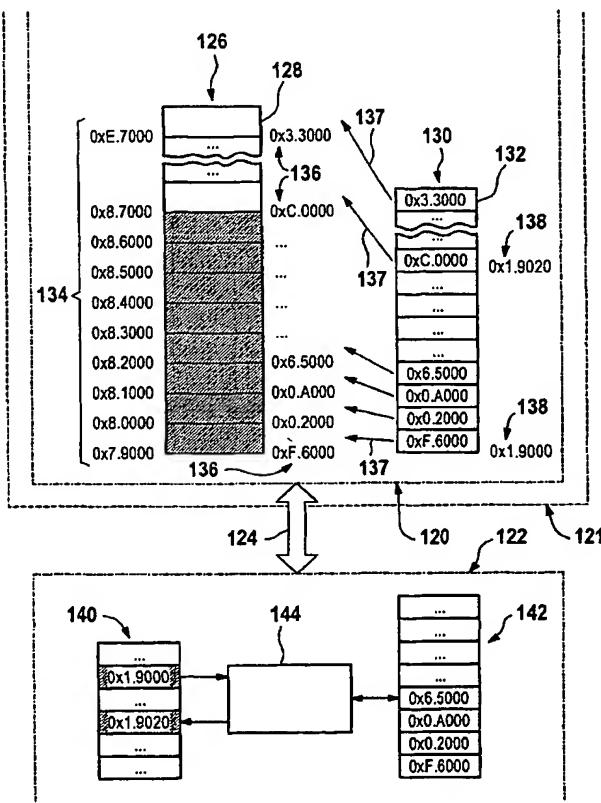
Veröffentlicht:

— mit internationalem Recherchenbericht

[Fortsetzung auf der nächsten Seite]

(54) Title: METHOD FOR TRANSFERRING DATA VIA A DATA BUS

(54) Bezeichnung: VERFAHREN ZUM ÜBERTRAGEN VON DATEN ÜBER EINEN DATENBUS



(57) **Abstract:** The invention relates to a method for transferring data via a data bus (124), in particular via a PCI bus (124). According to the method, a second additional memory area (130) is created in a main memory (120) of a processor, in which all physical addresses (136, 138) of the memory pages (128) are filed. During the data transfer, the required physical addresses are transferred independently.

(57) **Zusammenfassung:** Es ist ein Verfahren zum Übertragen von Daten über einen Datenbus (124), insbesondere über den PCI-Bus (124), beschrieben. Das erfindungsgemäße Verfahren sieht vor, dass in einem Hauptspeicher (120) einer Recheneinheit ein zweiter zusätzlicher Speicherbereich (130) angelegt wird, in den alle physikalischen Adressen (136, 138) der Speicherseiten (128) abgelegt werden. Während der Datenübertragung werden die benötigten physikalischen Adressen selbstständig übertragen.

WO 03/073301 A1

BEST AVAILABLE COPY

WO 03/073301 A1



Zur Erklärung der Zweibuchstaben-Codes und der anderen Abkürzungen wird auf die Erklärungen ("Guidance Notes on Codes and Abbreviations") am Anfang jeder regulären Ausgabe der PCT-Gazette verwiesen.

10 Verfahren zum Übertragen von Daten über einen Datenbus

Stand der Technik

15 Ein Standard Personal Computer (PC) zeichnet sich durch einfache Erweiterbarkeit aus, die beispielsweise über sogenannte Steckkarten erfolgen kann, die in Erweiterungssteckplätze auf der Hauptplatine eines PCs gesteckt werden. Im Laufe der Zeit hat sich für diese Erweiterungen der sogenannte PCI-Bus (PCI: Peripheral
20 Component Interconnect) durchgesetzt. Dieser definiert neben den mechanischen und elektrischen Eigenschaften des Datenbusses auch die Konfiguration der Busmitglieder.

25 Aus der US Patentschrift 57 765 008 ist ein Computer mit einer Hauptplatine bekannt, bei der Steckplätze für Erweiterungskarten vorgesehen sind. Diese Karten sind über den PCI-Bus mit der Hauptplatine verbunden.

30 In einem der Steckplätze ist eine sogenannte "riser card" vorgesehen, die wiederum eine Anzahl von Einsteckmöglichkeiten zur Verfügung stellt. Auf diese Weise können eine Vielzahl von Steckkarten in die Hauptplatine eingesteckt und somit vielfältige Funktionen realisiert werden.

Jede PCI-Erweiterungskarte benötigt zur Anbindung an den PCI-Bus eine PCI-Schnittstelle bzw. ein PCI-Interface. Dies kann entweder ein spezieller eigenständiger integrierter
5 Schaltkreis (PCI-Bridge) oder eine in einer Hardwarebeschreibungssprache implementierte PCI-Schnittstelle bzw. ein PCI-Interface (PCI-Core) sein. Letzteres kann mit zusätzlicher frei definierbarer Logik in programmierbare Logikbausteine, wie beispielsweise ein FGPA
10 (FGPA: Field Programmable Gate Array), eingebettet werden.

Beide Varianten der Anbindung an den PCI-Bus erlauben die Entwicklung von Erweiterungskarten, die den PCI-Bus über diesen Schnittstellenbaustein an eine lokale, selbst
15 definierbare Logik, beispielsweise einen lokalen proprietären Bus, anbindet. Bei Verwendung einer FGPA-basierten Lösung ist die lokale Logik absolut frei gestaltbar. Im Falle kommerzieller PCI-Bridges sind die lokalen Schnittstellen durch den Hersteller dieser
20 Schaltkreise definiert.

Beim Starten des PCs ist es nach erfolgter Selbstkonfiguration des PCI-Busses (Plug and Play)¹ notwendig, die eigene lokale Logik auf der
25 Erweiterungskarte zu initialisieren. Nach erfolgter Initialisierung kann die Karte ihren eigenen Betrieb aufnehmen, bspw. das Einlesen von Daten in den Rechner. Für die Kommunikation (Initialisierung, Datentransfer) zwischen Erweiterungskarte und dem Betriebssystem des PCs ist ein
30 Treiber zuständig. Dabei handelt es sich um ein Programm (Software), das speziell auf die spezifischen Eigenheiten der Erweiterungskarte abgestimmt ist und somit die Karte korrekt initialisieren kann. Darüber hinaus bildet der Treiber die Standardschnittstelle des Betriebssystems, die

für die Kommunikation mit Erweiterungskarten zur Verfügung steht, auf die Erweiterungskarte ab. Der Treiber leitet die Datenlese- und Schreibanfragen des Betriebssystems in einer geeigneten Form an die Erweiterungskarte weiter.

5

Neben Datentransferanfragen von seiten des Betriebssystem ist es auch möglich, daß die Erweiterungskarte dem PC signalisiert, daß diese Daten benötigt oder abliefern möchte. Die Signalisierung von seiten der Erweiterungskarte 10 geschieht durch sogenannte Unterbrechungsanforderungen (Interrupte), die den normalen sequenziellen Programmablauf des PCs unterbrechen. Bei einem von einer Erweiterungskarte ausgelösten Interrupt wird in eine speziell für diese Unterbrechung vom Treiber zur Verfügung gestellte 15 Softwareroutine gesprungen, die sog. Interrupt-Service-Routine (ISR).

Die ISR erkennt anhand des Interrupts die Art der Unterbrechungsanforderung und bedient diese, indem diese 20 bspw. Daten von der Karte ausliest und in den Hauptspeicher des PCs ablegt und so die ausgelesenen Daten den Anwendungsprogrammen zur Weiterverarbeitung zur Verfügung stellt. Der Code in der ISR sollte möglichst kurz und kompakt sein, um die Dauer der Unterbrechung zu minimieren 25 und somit die Leistungsfähigkeit des gesamten Systems nicht unnötig zu beeinträchtigen.

Neben der Möglichkeit, daß der Prozessor mit Hilfe des Treibers den Datentransport zwischen Hauptspeicher und einer 30 Erweiterungskarte selbst ausführt, gibt es auch noch einen direkten Speicherzugriff, den sog. DMA-Transfer (DMA; Direct Memory Access: direkter Speicherzugriff). Bei dem DMA-Transfer erfolgt ein direkter Zugriff der Erweiterungskarte auf den PC-Hauptspeicher. Hierzu wird der

Hauptspeicherbereich, aus dem die Daten für die Karte gelesen oder in den die Daten geschrieben werden sollen, durch seine Startadresse direkt der Karte mitgeteilt. Diese erlangt daraufhin die Kontrolle auf dem PCI-Bus (Master) 5 und überträgt die Daten zwischen dem über die Startadresse definierten Hauptspeicherbereich und der PCI-Erweiterungskarte selbstständig ohne Belastung des Prozessors. Nachdem der Datentransfer beendet ist, löst die Karte einen Interrupt aus, um dem Treiber die 10 erfolgreiche Datenübertragung mitzuteilen.

Moderne Betriebssysteme wie Windows NT/2000 oder auch Linux sind sog. Multitasking Betriebssysteme, was bedeutet, daß diese mehreren Anwendungen, die gleichzeitig bearbeitet 15 werden sollen, Prozessorzeit in Form von kleinen Zeitscheiben im Bereich von jeweils Millisekunden zur Verfügung stellen. Durch dieses Zeitscheibenverfahren entsteht für den Benutzer des PCs der Eindruck, daß alle aktiven Anwendungen gleichzeitig arbeiten. Da physisch bzw. 20 physikalisch unabhängig von der Anzahl der laufenden Anwendungen jedoch nur ein begrenzter Umfang an Hauptspeicher zur Verfügung steht, verwenden moderne Prozessoren das Konzept des virtuellen Speichers, um diese Beschränkung zu umgehen und allen Anwendungen den von ihnen 25 benötigten Speicher zur Verfügung zu stellen.

Das Konzept des virtuellen Speichers beruht darauf, sämtlichen Anwendungen einen sehr großen (virtuellen) Hauptspeicher vorzutäuschen (bspw. 2 GB), auch wenn 30 wesentlich weniger physischer Speicher zur Verfügung steht (z.B. 128 MB). Dies wird dadurch erreicht, daß der Speicher in viele kleine Einheiten, den sog. Seiten oder Pages unterteilt wird. Die typische Größe für solch eine Page beträgt 4.096 Byte. Fordert nun eine Anwendung

Speicher an (bspw. für Daten oder Programmcode), so wird dieser der benötigte Speicher in Form einer gewissen Anzahl von Seiten zur Verfügung gestellt. Die Einteilung des Speichers in Pages wird von allen modernen Prozessoren 5 durch Hardware unterstützt. Sobald eine Anwendung auf den Speicher zugreift, übersetzt eine sog. Memory Management Unit (MMU) die virtuellen Speicheradressen in physikalische Adressen. Läuft die Zeitscheibe einer Anwendung A ab und benötigt die nächste Anwendung B auch Speicher, der 10 ebenfalls gerade von der ersten Anwendung A benutzt wurde, so werden die Seiten der ersten Anwendung A ausgelagert, bspw. auf die Festplatte. Damit gehen die Daten oder der Code von der Anwendung A nicht verloren, aber Anwendung B kann dennoch den zuvor von Anwendung A genutzten 15 physikalischen Speicher nutzen.

Sollen größere Datenmengen, wie z.B. Videodaten, zwischen einer PCI-Erweiterungskarte und dem Hauptspeicher des PCs übertragen werden, so ist im Hauptspeicher des PCs ein 20 entsprechend großer Bereich für diese Daten zu reservieren. Ist der reservierte Bereich größer als eine Speicherseite (bspw. 4.096 Byte), so setzt sich dieser zwangsläufig aus mehreren Speicherseiten zusammen. Diese Seiten besitzen dann logisch aufeinanderfolgende virtuelle Adressen. Diese 25 virtuellen Adressen werden von der MMU auf den physikalisch vorhandenen Speicher abgebildet, wobei durch die Übersetzung die virtuellen Adressen nicht zwangsläufig auch auf physikalisch aufeinanderfolgende Adressen abgebildet werden müssen.

30

Die den virtuellen Seiten entsprechenden physikalischen Seiten, können somit weit verstreut im physikalischen Hauptspeicher des PCs liegen. Ein Datentransfer über den PCI-Bus arbeitet hingegen ausschließlich mit den

physikalischen Adressen des Speichers. Damit ein DMA-Transfer die im virtuellen Speicher zusammenhängenden Daten in der richtigen Reihenfolge liest bzw. die Daten so schreibt, daß sie anschließend zusammenhängend im 5 virtuellen Speicher liegen, muß dieser also auf die zufällig verteilten physikalischen Speicheradressen der einzelnen Seiten zugreifen. Dies bedeutet, daß immer nur 4.096 Bytes innerhalb eines Transfers gelesen oder geschrieben werden können.

10

Dieses Problem wird derzeit dadurch gelöst, daß der Treiber zu jeder virtuellen Speicherseite die physikalische Adresse ermittelt und der Erweiterungskarte zur Verfügung stellt. Nach dem heutigen Stand der Technik sind zwei Methoden 15 hierfür bekannt. Nach der ersten Methode enthält das PCI-Interface auf der Erweiterungskarte einen begrenzten Registerspeicher für die Seitenadressen des Hauptspeichers. Nach der zweiten Methode enthält das PCI-Interface auf der Erweiterungskarte genug Registerspeicher um alle 20 Seitenadressen für einen kompletten Datensatz aufnehmen zu können.

Als Beispiel zur Veranschaulichung wird im folgendem davon ausgegangen, daß ein Videobild mit CCIR-Auflösung von der 25 Erweiterungskarte in den Rechner übertragen werden soll.

Ein CCIR-Bild besteht aus 720 * 576 Bildpunkten. Jeder Bildpunkt benötigt im YCbCr-Format 2 Byte, woraus sich eine Datenmenge für ein CCIR-Bild von 30

$$720 * 576 * 2 \text{ Byte} = 829.440 \text{ Byte ergibt.}$$

Diese 829.440 Byte belegen

$$829.440 / 4.096 = 202,5,$$

also aufgerundet 203 Pages im Hauptspeicher des PCs.

5 Nach der ersten Methode enthält das PCI-Interface auf der Erweiterungskarte einen begrenzten (Register)-Speicher für die Seitenadressen des Hauptspeichers. Verfügt die Erweiterungskarte bspw. über acht Register für Speicheradressen ($8 * 32$ Bit = 32 Byte), müssen vor Beginn
10 eines Datentransfers diese acht Register mit gültigen Adressen initialisiert werden. Danach können maximal

$$8 * 4.096 = 32.786 \text{ Byte}$$

15 selbständig von der Karte per DMA in den Hauptspeicher des PCs übertragen werden. Nach jedem DMA-Transfer muß ein Interrupt ausgelöst werden, um den erfolgreichen Transfer bekannt zu geben und damit acht neue Speicheradressen anzufordern. Nimmt man beispielsweise an, daß 25 CCIR-
20 Bilder pro Sekunde in den Rechner übertragen werden sollen, so ergibt sich:

$$829.440 * 25 \text{ 1/s} = 20.736.000 \text{ Byte/s}$$

25 $20.736.000 \text{ Byte/s} / 32.768 \text{ Byte} = 632,8 \text{ 1/s}$

Folglich erhält man allein für die Übertragung einer Sekunde Bilddaten 633 Interrupte pro Sekunden.

30 Nach der zweiten Methode enthält das PCI-Interface auf der Erweiterungskarte genügend (Register)-Speicher, um alle Seitenadressen für ein komplettes CCIR-Bild aufnehmen zu können. Dies müssen für diesen Anwendungsfall also mindestens 203 Register mit je 32 Bit (812 Byte) sein.

Damit ist für jedes Bild nur ein Interrupt nötig, der den erfolgreichen Datentransfer bestätigt und für das nächste Bild wieder 203 (neue) Adressen anfordert. Damit ergeben sich bei 25 Bildern pro Sekunde auch 25 Interrupte pro

5 Sekunde.

Bei der vorstehend erläuterten ersten Methode wird die Performance des Systems aufgrund der großen Anzahl von Interruptanforderungen erheblich beeinträchtigt.

10 Jede dieser Anforderungen zwingt das System, die derzeitige Anwendung zu unterbrechen, den aktuellen Inhalt der CPU-Registerwerte zwischenzuspeichern, die Quelle für den Interrupt zu ermitteln und die entsprechende ISR auszuführen. Anschließend wird nach Wiederherstellung der 15 alten CPU-Registerwerte zu der zuvor unterbrochenen Anwendung zurückgekehrt.

Der Vorteil dieser Methode liegt bedingt durch die geringe Anzahl an zu speichernden Seitenadressen in dem geringen 20 Bedarf an Adreßregistern (Speicherplatz gleich Chipfläche bzw. FPGA-Ressourcen) im PCI-Interface auf der Erweiterungskarte.

Die zweite Methode bietet im Gegensatz zur ersten Methode 25 eine erheblich gesteigerte Performance, da das System wesentlich weniger Interruptanforderungen verarbeiten muß. Dieser Vorteil wird aber durch einen deutlichen Mehraufwand an Hardware erkauft. Der Hardwareaufwand (Chipfläche bzw. FPGA-Ressourcen) liegt im genannten Beispiel um Faktor 25 30 höher als derjenige bei der ersten Methode.

Vorteile der Erfindung

Das erfindungsgemäße Verfahren dient zum Übertragen von Daten über einen Datenbus, und zwar zwischen einer

5 Speichereinrichtung, die in Seiten bzw. Pages unterteilt ist, wobei auf die Seiten mittels physikalischer Adressen zugegriffen werden kann, und einer mit der Speichereinrichtung über den Datenbus verbundenen elektronischen Einheit.

10

Die Speichereinrichtung umfaßt erfindungsgemäß einen ersten und einen zweiten Speicherbereich, wobei der erste Speicherbereich zur Speicherung von Daten vorgesehen ist und der zweite Speicherbereich die physikalischen Adressen 15 der Seiten des ersten Speicherbereichs oder auch der gesamten Speichereinrichtung enthält. Die Reihenfolge der physikalischen Adressen im zweiten Speicherbereich entspricht dabei der Reihenfolge der virtuellen Seiten des ersten Speicherbereichs.

20

Während der Datenübertragung werden die benötigten physikalischen Adressen aus dem zweiten Speicherbereich selbständig auf die elektronische Einheit übertragen. Damit ist der elektronischen Einheit die Lage der physikalischen Adressen bekannt. Selbständig übertragen bedeutet eine Übertragung ohne Eingriff eines Systemprozessors, der der Speichereinrichtung zugeordnet ist. Damit erweitert das Verfahren den Ansatz, daß die elektronische Einheit selbständig und ohne Eingriff des Systemprozessors Daten 25 aus der Speichereinrichtung übertragen kann nun auch auf die selbständige Übertragung von Adreßdaten.

30 Das erfindungsgemäße Verfahren verbindet somit die Vorteile der beiden vorstehend beschriebenen Methoden, nämlich

geringer Speicherbedarf und geringe Systembelastung, und vermeidet deren Nachteile. Somit ermöglicht das Verfahren je nach bisher verwendeter Datenübertragungsmethode entweder eine wesentliche Verringerung der Anzahl der 5 Interruptanforderungen und damit eine erhebliche Erhöhung der System-Performance oder im anderen Fall einen stark verringerten Bedarf an Hardwareressourcen, die für die Realisierung des PCI-Interfaces notwendig sind. Dabei erfährt die Art der eigentlichen Datenübertragung keine 10 Änderung und wird somit nicht beeinflußt. Es wird lediglich die Art und Weise der Übertragung der Seitenadressen geändert.

Mit dem erfindungsgemäßen Verfahren ist es somit zum einen 15 nicht mehr notwendig, alle für eine vollständige Datenübertragung notwendigen Adressen in einem Speicher auf der elektronischen Einheit zu halten. Der Nachteil der zweiten Methode, nämlich der hohe Speicherbedarf innerhalb 20 der Schnittstelle der elektronischen Einheit ist damit aufgehoben. Andererseits wird, obwohl die elektronische Einheit nicht mehr alle physikalischen Adressen speichert, nicht mehr, wie bei der ersten Methode, nach einer gewissen 25 Anzahl abgearbeiteter Adressen eine Interruptanforderung an das System gesendet und damit eine Unterbrechung der laufenden Anwendung ausgelöst.

Das erfindungsgemäße Verfahren beinhaltet Änderungen in der Hard- sowie der Softwareimplementierung bei PCI-Interface und Treiber. Die Hardwareimplementierung ist sowohl bei 30 PCI-Bridges in Standard-IC-Technik als auch bei PCI-Cores in einer Hardwarebeschreibungssprache anwendbar. Der Softwareteil ist unabhängig von dem verwendeten Betriebssystem und kann in jeder Treiberimplementierung eingesetzt werden.

Vorteilhafterweise wird zu Beginn der Datenübertragung eine Startadresse des zweiten Speicherbereichs an die elektronische Einheit übergeben. Damit sind der 5 elektronischen Einheit die Lage, also der physikalische Ort, der physikalischen Adressen der Seiten der Speichereinrichtung bekannt.

Das erfindungsgemäße Verfahren eignet sich sowohl zum 10 Schreiben von Daten von der elektronischen Einheit in die Speichereinrichtung als auch zum Lesen von in der Speichereinrichtung abgelegten Daten.

Als Datenbus bietet sich der PCI-Bus an, der insbesondere 15 bei PCs weit verbreitet ist.

In Ausgestaltung des erfindungsgemäßen Verfahrens ist die Speichereinrichtung ein auf einer Hauptplatine einer elektronischen Recheneinheit (Computer) angeordneter 20 Hauptspeicher. Als elektronische Einheit ist eine in einem Einstechplatz der Hauptplatine eingesetzte Steckkarte bzw. Erweiterungskarte vorgesehen. In diesem Fall ist die CPU der elektronischen Recheneinheit der Systemprozessor.

25 Typischerweise werden die physikalischen Adressen per DMA-Transfer auf die elektronische Einheit übertragen.

Die erfindungsgemäße Speichereinrichtung ist in Seiten unterteilt und umfaßt einen ersten Speicherbereich und 30 einen zweiten Speicherbereich. Der erste Speicherbereich ist für Daten vorgesehen, d.h. in diesem Speicherbereich sind die zu lesenden Daten abgespeichert bzw. in diesen Speicherbereich werden Daten geschrieben. In dem zweiten Speicherbereich sind die physikalischen Adressen der Seiten

des ersten Speicherbereichs abgelegt. Diese Adressen erlauben entweder den Zugriff auf die in dem ersten Speicherbereich abgelegten Daten und/oder ermöglichen ein Beschreiben des ersten Speicherbereichs.

5

Die erfindungsgemäße Speichereinrichtung wird vorzugsweise als ein auf einer Hauptplatine einer elektronischen Recheneinheit angeordneter Hauptspeicher verwendet.

10 Die erfindungsgemäße Hauptplatine einer elektronischen Recheneinheit (Computer) zeichnet sich dadurch aus, daß als Hauptspeicher eine erfindungsgemäße Speichereinrichtung dient.

15 Die erfindungsgemäße elektronische Recheneinheit weist einen Hauptspeicher auf. Als Hauptspeicher ist eine erfindungsgemäße Speichereinrichtung vorgesehen. Die CPU der Recheneinheit dient als Systemprozessor. Der Hauptspeicher ist vorzugsweise auf einer Hauptplatine 20 angeordnet.

Das erfindungsgemäße System umfaßt eine erfindungsgemäße Speichereinrichtung und eine erfindungsgemäße elektronische Einheit. Diese sind über einen Bus, vorzugsweise den PCI- 25 Bus, miteinander zum Übertragen von Daten und Adressen verbunden.

Die erfindungsgemäße elektronische Einrichtung, bei der eine elektronische Recheneinrichtung bzw. eine CPU, eine 30 Speichereinrichtung und eine elektronische Einheit in einem Bauteil integriert sind, stellt ein kompaktes System dar. Bei diesem sind bspw. auf einer Kante oder sogar in einem einzelnen Chip (SoC: System on Chip) alle Komponenten integriert.

Das erfindungsgemäße Computerprogramm weist Programmcodemittel auf, um alle Schritte eines vorstehend beschriebenen Verfahrens durchzuführen. Das Computerprogramm wird auf einem Computer oder einer

5 Recheneinheit, insbesondere einer erfindungsgemäßen elektronischen Recheneinheit, ausgeführt.

Das erfindungsgemäße Computerprogrammprodukt umfaßt eben diese Programmcodemittel und ist auf einem computerlesbaren

10 Datenträger gespeichert.

Weitere Vorteile und Ausgestaltungen der Erfindung ergeben sich aus der Beschreibung und der beiliegenden Zeichnung.

15 Es versteht sich, daß die vorstehend genannten und die nachstehend noch zu erläuternden Merkmale nicht nur in der jeweils angegebenen Kombination, sondern auch in anderen Kombinationen oder in Alleinstellung verwendbar sind, ohne den Rahmen der vorliegenden Erfindung zu verlassen.

20

Zeichnung

Die Erfindung ist anhand von Ausführungsbeispielen in der Zeichnung dargestellt und wird im folgenden unter

25 Bezugnahme auf die Zeichnung näher erläutert.

Figur 1 zeigt die Grundstruktur eines PCI-Systems in schematischer Darstellung.

30 Figur 2 zeigt eine Erweiterungskarte mit einem PCI-Bus-Interface.

Figur 3 verdeutlicht den Zusammenhang zwischen virtuellen und physikalischen Adressen.

Figur 4 zeigt eine bevorzugte Ausführungsform der erfindungsgemäßen elektronischen Recheneinheit in schematischer Darstellung.

5

Figur 5 veranschaulicht das Zusammenwirken einer erfindungsgemäßen Speichereinrichtung mit einer Erweiterungskarte.

10 Figur 6 zeigt eine bevorzugte Ausführungsform des erfindungsgemäßen Verfahrens in einem Flußdiagramm.

15 In Figur 1 ist die Grundstruktur eines PCI-Systems, insgesamt mit der Bezugsziffer 10 bezeichnet, dargestellt.

Zu erkennen ist eine CPU 12, eine sogenannte Host-Bridge 14, ein Hauptspeicher 16, eine erste PCI-Karte 18, eine zweite PCI-Karte 20 und ein PCI-Bus 22.

20

Die erste PCI-Karte 18 umfaßt ein erstes PCI-Interface 24 und eine erste lokale Logik 26. Die zweite PCI-Karte 20 weist entsprechend ein zweites PCI-Interface 28 und eine zweite lokale Logik 30 auf.

25

Die PCI-Karten 18 und 20 sind mit den Schnittstellen, den PCI-Interfaces 24 und 28, an den PCI-Bus 22 angeschlossen. Damit sind auch die lokalen Logiken 26 und 30 an den PCI-Bus 22 angeschlossen.

30

Die Anbindung der CPU 12 und des Hauptspeichers 16 erfolgt über die Host-Bridge 14.

In Figur 2 ist eine Erweiterungs- bzw. Einstekkkarte 40 mit einem Slotblech 42 dargestellt.

Auf der Einstekkkarte 40 befindet sich ein PCI-Block 44,

5 der wiederum eine lokale Logik 46 und ein PCI-Interface 48 umfaßt. Wie durch einen Doppelpfeil 50 verdeutlicht ist, ist das PCI-Interface 48 mit dem PCI-Bus 52 der Hauptplatine der elektronischen Recheneinheit verbunden.

10 Auf der Einstekkkarte 40 sind weiterhin ein erster Logikblock 54, ein zweiter Logikblock 56 und ein dritter Logikblock 58 vorgesehen. Des weiteren sind Steckverbinder 60 dargestellt, die den festen Halt der Einstekkkarte 40 sichern und Anschlüsse für einen Datenaustausch umfassen.

15 Die lokale Logik 46 ist über einen lokalen proprietären Bus 62 mit dem ersten und dem zweiten Logikblock 54 und 56 verbunden. Zwischen der lokalen Logik 46 und dem dritten Logikblock 58 besteht auch eine direkte Anbindung 64.

20 In Figur 3 ist der Zusammenhang zwischen virtuellen und physikalischen Adressen verdeutlicht.

Dargestellt ist die virtuelle Organisation 70 des Speichers, nachfolgend als virtueller Speicher bezeichnet, und die physikalische Organisation 72 des Speichers, nachfolgend als physikalischer Speicher bezeichnet. In dem virtuellen Speicher 70 ist ein Datenelement 74 enthalten. Die mit der Bezugsziffer 76 bezeichneten Angaben 0x14, 0x13 30 und 0x12 sind Beispiele für virtuelle Adressen. Beispiele für physikalische Adressen 0xA9, 0x7D und 0x05 sind mit der Bezugsziffer 78 bezeichnet.

Das Datenelement 74 in dem physikalischen Speicher 70 umfaßt drei Seiten 82, die mit den virtuellen Adressen 76 gekennzeichnet sind. Diese Seiten 82 besitzen logisch aufeinanderfolgende virtuelle Adressen. Mit der 5 Bezugsziffer 84 sind drei physikalische Seiten im physikalischen Speicher 72 bezeichnet. Pfeile 86 verdeutlichen die Zuweisung von dem virtuellen Speicher 70 zu dem physikalischen Speicher 72. Zu erkennen ist, daß die den virtuellen Seiten 82 entsprechenden physikalischen 10 Seiten 84 weit verstreut im Hauptspeicher liegen. Die virtuellen Adressen 76 werden von der MMU auf den physikalisch vorhandenen Speicher 72 abgebildet.

Der Datentransfer auf dem PCI-Bus arbeitet ausschließlich 15 mit den physikalischen Adressen 78 des physikalischen Speichers 72. Damit ein DMA-Transfer die im virtuellen Speicher 70 zusammenhängenden Daten in der richtigen Reihenfolge liest bzw. die Daten so schreibt, daß diese anschließend zusammenhängend im virtuellen Speicher 70 20 liegen, muß im Rahmen des Transfers auf die zufällig verteilten physikalischen Speicheradressen 78 zugegriffen werden.

Figur 4 zeigt eine bevorzugte Ausführungsform einer 25 erfindungsgemäßen elektronischen Recheneinheit, insgesamt mit der Bezugsziffer 100 bezeichnet. Die Recheneinheit 100 ist bspw. ein Computer bzw. ein PC. Zu erkennen ist eine Hauptplatine 102, auf der eine CPU 104 und eine Speichereinrichtung 106, der Hauptspeicher 106 der 30 elektronischen Recheneinheit 100, angeordnet sind. Die CPU 104 und der Hauptspeicher 106 sind über einen Standardbus 108 miteinander verbunden.

Der Hauptspeicher 106 umfaßt einen ersten Speicherbereich 110 und einen zweiten Speicherbereich 112. Der erste Speicherbereich 110 ist für Daten vorgesehen. Dies bedeutet, daß in diesem Bereich die Daten abgelegt sind, 5 auf die zugegriffen werden soll und/oder in den ersten Speicherbereich 110 Daten, bspw. von der Einstekkarte 40, geschrieben werden können. Der zweite Speicherbereich 112 enthält die physikalischen Adressen 78 der Seiten des ersten Speicherbereichs 110 bzw. des gesamten 10 Hauptspeichers 106.

In Figur 5 ist das Zusammenspiel zwischen einer elektronischen Recheneinheit 121 und einer Erweiterungskarte 122, die über einen PCI-Bus 124 verbunden 15 sind, veranschaulicht. Die elektronische Recheneinheit 121 enthält einen Hauptspeicher 120.

Mit der Bezugsziffer 126 ist der in Seiten 128 unterteilte erste Speicherbereich des Hauptspeichers 120 dargestellt. 20 Eine Seite 128 umfaßt typischerweise 4.096 Byte.

Mit der Bezugsziffer 130 ist der zweite Speicherbereich, nämlich der Speicherbereich für die Adreßtabelle, gekennzeichnet, der in Adreßspeicherplätze 132 unterteilt 25 ist.

Die mit 134 bezeichneten Angaben (0xE.7000 usw.) sind die logisch aufeinanderfolgenden virtuellen Adressen. Angaben 136 (0x3.3000 usw.) sind die beliebig verteilten 30 physikalischen Adressen der Seiten 128 des ersten Speicherbereichs 126. Pfeile 137 verdeutlichen die Zuordnung der Daten der in dem zweiten Speicherbereich 130 enthaltenen Adreßtabelle auf die physikalischen Adressen 136 der Seiten 128 des ersten Speicherbereichs 126. D.h. in

dem zweiten Speicherbereich 130 sind in einer Adreßtabelle die physikalischen Adressen 136 des ersten Speicherbereichs 126 enthalten.

5 Die mit 138 gekennzeichneten Angaben 0x1.9020 und 0x1.9000 sind die physikalischen Adressen der Adreßspeicherplätze 132 des zweiten Speicherbereichs 130, wobei die Angabe 0x1.9000 die physikalische Basisadresse, d.h. die benötigte Startadresse, der Adreßtabelle wiedergibt.

10

In der PCI-Erweiterungskarte 122 ist ein Registerspeicher 140 vorgesehen, der die lokalen Register der Erweiterungskarte 122 enthält. In diesen Registern sind die benötigten physikalischen Adressen 138 des zweiten Speicherbereichs 130 abgelegt. Die Angabe 0x1.9000 stellt die Basisadresse der Adreßtabelle dar. Die Angabe 0x1.9020 ist die beispielhafte Basisadresse für den nächsten Adreßblock aus der Adreßtabelle bei einer beispielhaften Adreßblocklänge von acht.

15

In einem Adreßspeicher 142 sind lokale Zieladressen, d.h. der Inhalt des zweiten Speicherbereichs 130 bzw. die physikalischen Adressen 136 des ersten Speicherbereichs 126, enthalten. Die Tiefe beträgt im Beispiel acht Register, die per PCI-Master-DMA durch die PCI-Erweiterungskarte 122 gefüllt wird. Der Adreßspeicher 142 enthält somit Adressen von acht Seiten 128. Zwischen dem Registerspeicher 140 und dem Adreßspeicher 142 ist eine Adreßverwaltungslogik 144 vorgesehen.

20

Bei der Initialisierung legt der Treiber im Hauptspeicher des PCs einen ersten Speicherbereich 126 mit der von den Bilddaten benötigten Größe an. Anhand dieser Größe berechnet dieser die Anzahl der benötigten Speicherseiten

128, die der erste Speicherbereich 126 einnimmt. Davon ausgehend wird der zweite Speicherbereich 130 angelegt, die physikalischen Adressen 136 des ersten Speicherbereichs 126 werden ermittelt (in Figur 5 0xF.6000, 0x0.2000 usw.) und 5 im zweiten Speicherbereich 130 abgelegt. Die physikalische Basisadresse, die Startadresse, der Adreßtabelle im zweiten Speicherbereich 130 (in diesem Fall 0x1.9000) teilt der Treiber der Erweiterungskarte mit. Diese speichert den Wert in einem lokalen Register, in diesem Fall in dem 10 Registerspeicher 140.

Beim Start der Datenübertragung beginnt die Karte, ausgehend von der Basisadresse 138 der Adreßtabelle (0x1.9000) bspw. acht Basisadressen 136 der Speicherseiten 15 128 des für die Datenübertragung reservierten Bereichs (0xF.6000, 0x0.2000 usw.) aus dem Hauptspeicher 120 des PCs per PCI-Master-DMA auszulesen und lokal in 142 abzulegen. Im Anschluß daran werden die Daten ebenfalls per PCI-Master-DMA in die auf diese Weise referenzierten 20 Speicherseiten 128 übertragen. Sobald 8 * 4.096 Datenbyte übertragen sind, holt sich die Karte in einem weiteren PCI-Master-DMA-Zyklus die neuen Basisadressen, d.h. die physikalischen Adressen 136, für die nächsten acht Speicherseiten.

25 Die interne Logik des PCI-Controllers muß zusätzlich zu der herkömmlichen Logik weitere Register umfassen, die der Verwaltung der Speicheradressen für die Adreßtabelle dienen. Ähnlich wie bei der Verwaltung der Adressen für den 30 Datenspeicher wird auch hier die Adresse zwischengespeichert (0x1.9020), deren Speicherzelle wiederum die Adresse enthält (0x0.C000), die auf die nächste zu nutzende Speicherseite verweist. Anstatt eine Interruptanforderung gemäß der bekannten ersten Methode

auszulösen, werden nun die lokalen Adreßregister selbständig vom PCI-Controller aktualisiert.

Durch das erfindungsgemäße Verfahren ist es möglich, daß 5 trotz der geringeren Anzahl von Zieladreßregistern und den damit gesparten Hardwareressourcen nur ein Interrupt pro Datenblock ausgelöst wird. Dieser eine Interrupt ist auch weitestgehend unabhängig von der Größe des Datenblocks. Im Falle von Datenblöcken, die größer als 4 MegaByte sind, 10 sind gegebenenfalls mehrere Adreßtabellen zu verwenden, da hierbei die Adreßtabelle selbst mehr als eine Speicherseite einnimmt, oder es wird über mehr als eine Interruptanforderung pro Datenblock die Aktualisierung der Adreßtabelle angefordert.

15

In Figur 6 ist in einem Flußdiagramm eine bevorzugte Ausführungsform des erfindungsgemäßen Verfahrens wiedergegeben. Mit der Bezugsziffer 150 ist auf die Verfahrensschritte hingewiesen, die durch den Treiber 20 ausgeführt werden. Bezugsziffer 152 verweist auf die in der Erweiterungskarte durchgeföhrten Verfahrensschritte.

In einem Schritt 154 wird der erste Speicherbereich, der Datenspeicher, und in einem Schritt 156 die Adreßtabelle 25 bzw. der zweite Speicherbereich bestimmt. Anschließend werden in einem Schritt 158 die physikalischen Adressen des Datenspeichers ermittelt und in der Adreßtabelle abgelegt.

Es erfolgt, wie mit einem Pfeil 160 verdeutlicht, die 30 Initialisierung der Erweiterungskarte und die Übertragung der Startadresse der Adreßtabelle.

Beim Start der Datenübertragung, wie mit einem Pfeil 162 gezeigt, wird zunächst der erste Teil der Adreßtabelle in

einem Schritt 164 aus dem PC-Hauptspeicher in den lokalen Speicher der Erweiterungskarte eingelesen. Anschließend erfolgt in einem Schritt 166 das Übertragen der (Nutz)-Daten der Speicherseite $x + 0$. Ist der Datenblock nicht 5 bereits vollständig übertragen, dann folgt in einem Schritt 168 das Übertragen der Daten der Seite $x + 1$. Diese Vorgänge wiederholen sich 170, bis in einem Schritt 172 das Übertragen der Daten der Speicherseite $x + 7$ durchgeführt wird. Die Übertragung erfolgt, wie mit gestrichelten 10 Doppelpfeilen 174 veranschaulicht per PCI-Master-DMA.

In einem Schritt 176 wird überprüft, ob der zu übertragene Datenblock zu Ende ist, d.h. bereits vollständig übertragen wurde. Ist dies nicht der Fall springt der Ablauf gemäß 15 Pfeil 178 zurück zum Verfahrensschritt 164. Ist der Datenblock vollständig übertragen, erfolgt, wie mit einem Pfeil 180 verdeutlicht, ein Interrupt.

Nachfolgend werden in einem Schritt 182 empfangene Daten 20 verarbeitet bzw. neue Daten zum Senden vorbereitet.

Die PCI-Erweiterungskarte ist in der Lage als Bus-Master, selbständig und ohne Eingriff des Systemprozessors Daten per DMA aus dem Hauptspeicher des PCs zu lesen oder in 25 diesen zu schreiben. Hierzu benötigt die Erweiterungskarte jedoch die physikalischen Adressen der Hauptspeicherseite, aus denen gelesen bzw. in die geschrieben werden soll. Bei den zum Stand der Technik beschriebenen zwei Methoden werden diese Adressen vom Treiber und damit vom 30 Systemprozessor bzw. von der CPU aktiv in den Registerspeicher des PCI-Interfaces geschrieben.

Das erfindungsgemäße Verfahren erweitert den Ansatz, nach dem die Erweiterungskarte selbständig Daten per DMA ohne

Eingriff der CPU übertragen kann, nun auch auf die selbständige Übertragung von Adressen. Neben dem ersten Speicherbereich im Hauptspeicher des PCs, der als Datenspeicher dient, wird noch ein zweiter, in der Regel 5 kleinerer Speicherbereich angelegt. In diesen zweiten Speicherbereich legt der Treiber nun alle physikalischen Adressen der Speicherseiten ab, aus denen der erste Speicherbereich besteht.

- 10 Bei Beginn der Datenübertragung wird dem PCI-Interface auf der Erweiterungskarte jetzt nur noch die Startadresse für den zweiten Speicherbereich übergeben. Damit ist der Erweiterungskarte die Lage der physikalischen Adressen der Hauptspeicherseiten bekannt. Während der Datenübertragung 15 kann sich die Erweiterungskarte nun die Ziel- oder Quelladressen für die Daten im ersten Speicherbereich je nach Bedarf aus dem zweiten Speicherbereich selbständig holen. Damit ist es einerseits nicht mehr notwendig, alle für eine komplette Datenübertragung benötigten Adressen im 20 Speicher auf der Erweiterungskarte zu halten. Der Nachteil der beschriebenen zweiten Methode, nämlich der hohe Speicherbedarf innerhalb des PCI-Interfaces auf der Erweiterungskarte ist damit aufgehoben. Andererseits wird nicht mehr, obwohl die Erweiterungskarte nicht mehr alle 25 physikalischen Adressen speichert, bspw. nach acht abgearbeiteten Adressen, wie bei der ersten Methode, eine Interruptanforderung an das System gesendet und damit eine Unterbrechung der laufenden Anwendung ausgelöst.
- 30 Die Karte liest die nächsten Seitenadressen selbst per DMA aus dem zweiten Speicherbereich in die internen Adreßregister. Damit wird nur noch für jeden Datenblock ein Interrupt ausgelöst. Die hohe Anzahl von Interrupten nach der ersten Methode wird demzufolge vermieden. Da innerhalb

der ISR nun auch nicht mehr die Seitenadressen übertragen werden müssen, kann die Zeit für die Abarbeitung der Routine ebenfalls deutlich verringert werden.

5 Der zweite Speicherbereich für die Adressen sollte dabei möglichst die Größe einer Seite, also bspw. 4.096 Byte, nicht überschreiten, da sonst auch für die Adressen, die in dem zweiten Speicherbereich gehalten werden, mehrere Seitenadressen notwendig wären. Dies ist jedoch im

10 allgemeinen völlig ausreichend, wie die folgende Beispielrechnung zeigt.

Bei modernen Prozessoren sind Adressen 32 Bit, also 4 Byte, groß. Damit ist es möglich, in einer Speicherseite

15 $4.096 \text{ Byte} / 4 \text{ Byte} = 1024$

(Seiten)-Adressen abzulegen. Mit diesen lassen sich 1024 Seiten ansprechen. Der auf diese Weise adressierbare

20 Speicherbereich umfaßt folglich:

$$1.024 * 4.096 \text{ Byte} = 4.194.304 \text{ Byte} = 4 \text{ MegaByte.}$$

Unabhängig davon ist das vorgestellte Konzept aber auch für

25 mehr als eine solche Adreßtabelle anwendbar. Somit sind auch Übertragungen umfangreicherer Datenmengen denkbar. In diesem Fall sind mehrere Speicherseiten als Adreßtabelle zu definieren und deren Anfangsadresse ist im PCI-Interface abzulegen.

Ansprüche

10 1. Verfahren zum Übertragen von Daten über einen Datenbus (22, 52, 124), zwischen einer Speichereinrichtung (16, 106, 120), die in Seiten (82, 84, 128) unterteilt ist und einen ersten Speicherbereich (110, 126) und einen zweiten Speicherbereich (112, 130) umfaßt, wobei auf die Seiten (82, 84, 128) mittels physikalischer Adressen (78, 136, 138) zugegriffen werden kann, und einer mit der Speichereinrichtung (16, 106, 120) über den Datenbus (22, 52, 124) verbundenen elektronischen Einheit (40, 122) und wobei der erste Speicherbereich (110, 126) zur Speicherung von Daten vorgesehen ist und der zweite Speicherbereich (112, 130) die physikalischen Adressen (78, 136, 138) der Seiten (82, 84, 128) des ersten Speicherbereichs (110, 126) enthält, bei dem während der Datenübertragung die benötigten physikalischen Adressen (78, 136, 138) aus dem zweiten Speicherbereich (112, 130) selbstständig auf die elektronische Einheit (40, 122) übertragen werden.

2. Verfahren nach Anspruch 1, bei dem zu Beginn der Datenübertragung eine Startadresse (138) des zweiten Speicherbereichs (112, 130) an die elektronische Einheit (40, 122) übergeben wird.

3. Verfahren nach Anspruch 1 oder 2, bei dem bei der Datenübertragung Daten von der elektronischen Einheit (40,

122) in die Speichereinrichtung (16, 106, 120) geschrieben werden.

4. Verfahren nach Anspruch 1 oder 2, bei dem bei der 5 Datenübertragung in der Speichereinrichtung (16, 106, 120) abgelegte Daten von der elektronischen Einheit (40, 122) gelesen werden.

5. Verfahren nach einem der Ansprüche 1 bis 4, bei dem der 10 Datenbus (22, 52, 124) ein PCI-Bus (22, 52, 124) ist.

6. Verfahren nach einem der Ansprüche 1 bis 5, bei dem die Speichereinrichtung (16, 106, 120) ein auf einer 15 Hauptplatine (110) einer elektronischen Recheneinheit (100, 121) angeordneter Hauptspeicher (16, 106, 120) ist und als elektronische Einheit (40, 122) eine in einem Einstckplatz 20 (60) der Hauptplatine (110) eingesetzte Steckkarte (40, 122) vorgesehen ist.

7. Verfahren nach einem der Ansprüche 1 bis 6, bei dem die 25 physikalischen Adressen (78, 136, 138) per DMA-Transfer auf die elektronische Einheit (40, 122) übertragen werden.

8. Speichereinrichtung, die in Seiten (82, 84, 128, 132) 25 unterteilt ist und einen ersten Speicherbereich (110, 126) und einen zweiten Speicherbereich (112, 130) umfaßt, wobei der erste Speicherbereich (110, 126) für Daten vorgesehen ist und in dem zweiten Speicherbereich (112, 130) die 30 physikalischen Adressen (78, 136, 138) der Seiten (82, 84, 128) des ersten Speicherbereichs (110, 126) abgelegt sind.

9. Verwendung einer Speichereinrichtung (16, 106, 120) nach 35 Anspruch 8 als ein auf einer Hauptplatine (110) einer

elektronischen Recheneinheit (100) angeordneter
Hauptspeicher (16, 106, 120).

10. Hauptplatine einer elektronischen Recheneinheit (100,
5 121), bei der als Hauptspeicher (16, 106, 120) eine
Speichereinrichtung (16, 106, 120) nach Anspruch 8 dient.

11. Elektronische Recheneinheit mit einer Hauptplatine
(110) und einem auf der Hauptplatine (110) angeordneten
10 Hauptspeicher (16, 106, 120), wobei als Hauptspeicher (16,
106, 120) eine Speichereinrichtung (16, 106, 120) nach
Anspruch 8 vorgesehen ist.

12. System mit einer Speichereinrichtung (16, 106, 120)
15 nach Anspruch 8 und einer elektronischen Einheit (40, 122),
die über einen Datenbus (22, 52, 124) miteinander verbunden
sind.

13. Elektronische Einrichtung, bei der eine elektronische
20 Recheneinrichtung, eine Speichereinrichtung nach Anspruch 8
und eine elektronische Einheit in einem Bauteil integriert
sind.

14. Computerprogramm mit Programmcodemitteln, um alle
25 Schritte eines Verfahrens nach einem der Ansprüche 1 bis 7
durchzuführen, wenn das Computerprogramm auf einem Computer
oder einer entsprechenden Recheneinheit (100, 121),
insbesondere einer elektronischen Recheneinheit (100, 121)
nach Anspruch 11, ausgeführt wird.

30 15. Computerprogrammprodukt mit Programmcodemitteln, die
auf einem computerlesbaren Datenträger gespeichert sind, um
ein Verfahren nach einem der Ansprüche 1 bis 7
durchzuführen, wenn das Computerprogramm auf einem Computer

oder auf einer entsprechenden Recheneinheit (100, 121), insbesondere einer elektronischen Recheneinheit (100, 121) nach Anspruch 11, ausgeführt wird.

1 / 4

FIG. 1

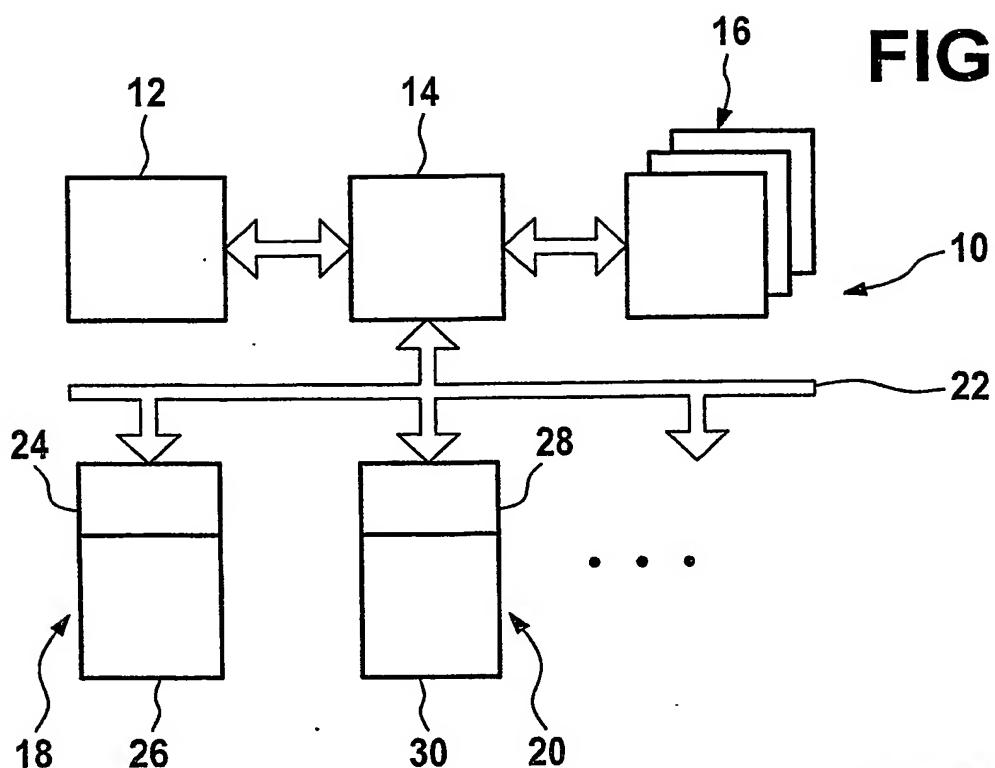
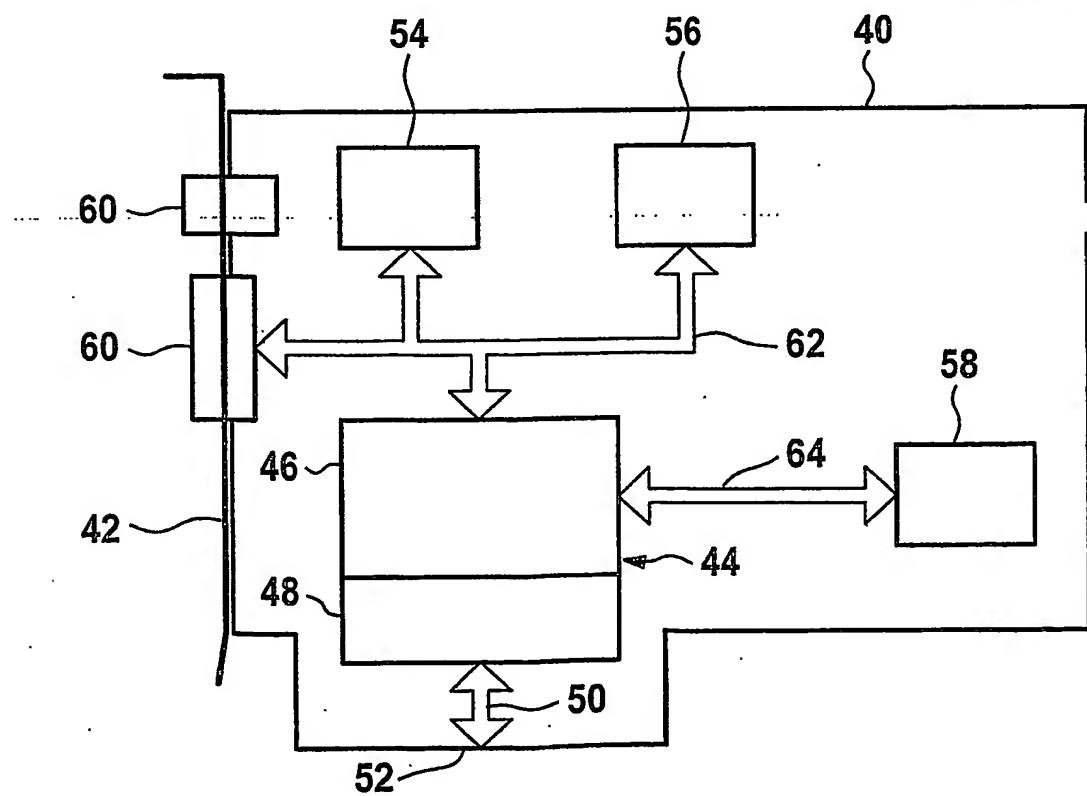


FIG. 2



2 / 4

FIG. 3

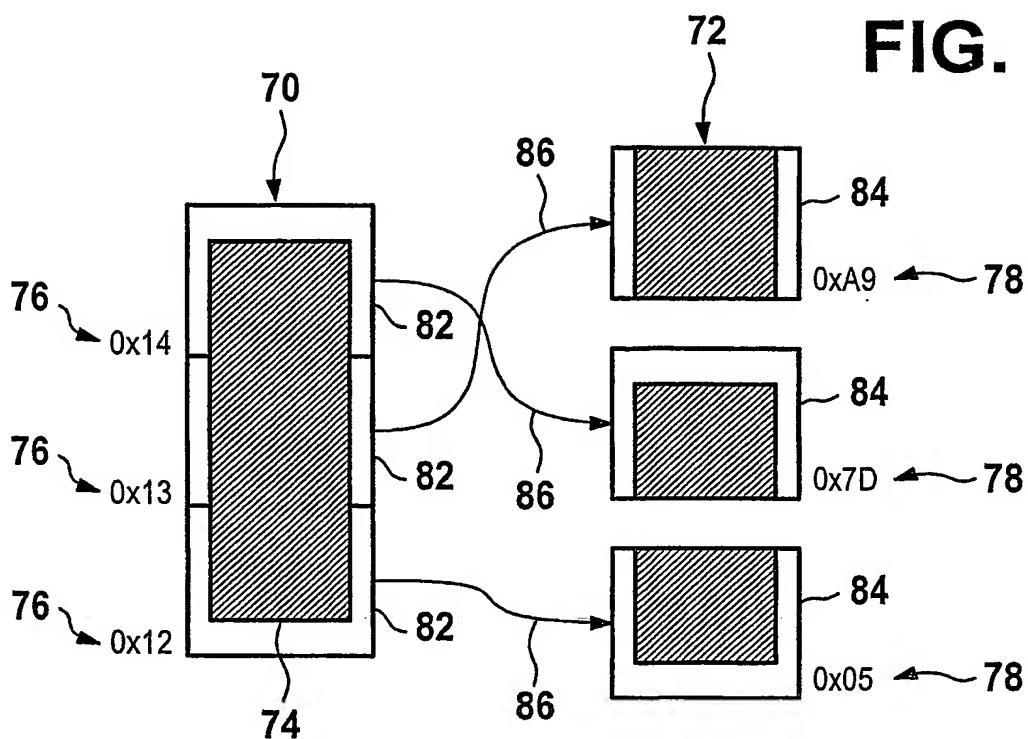


FIG. 4

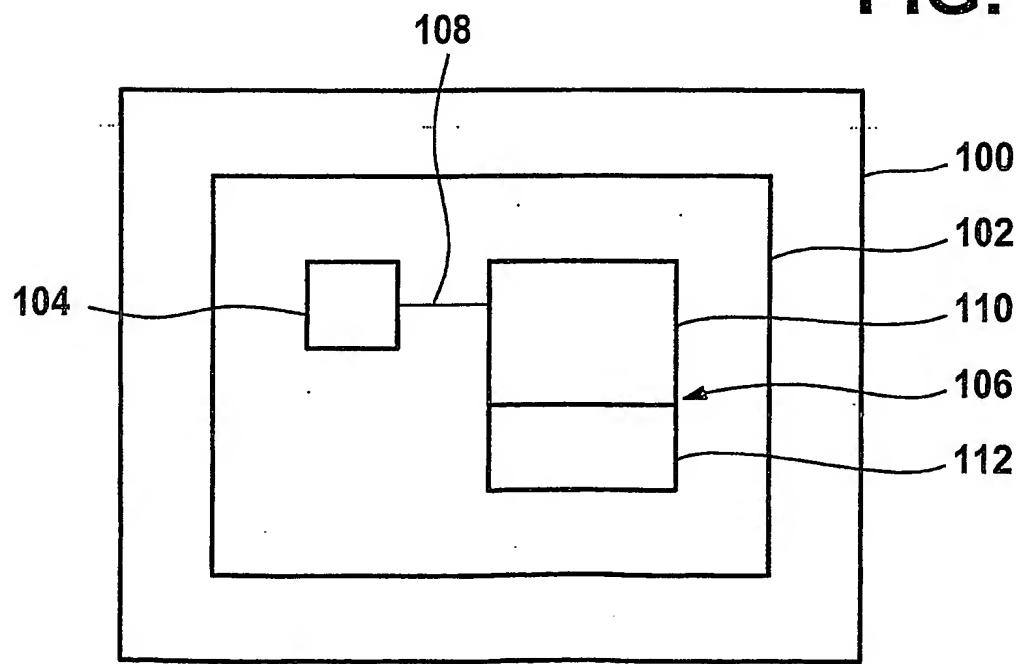
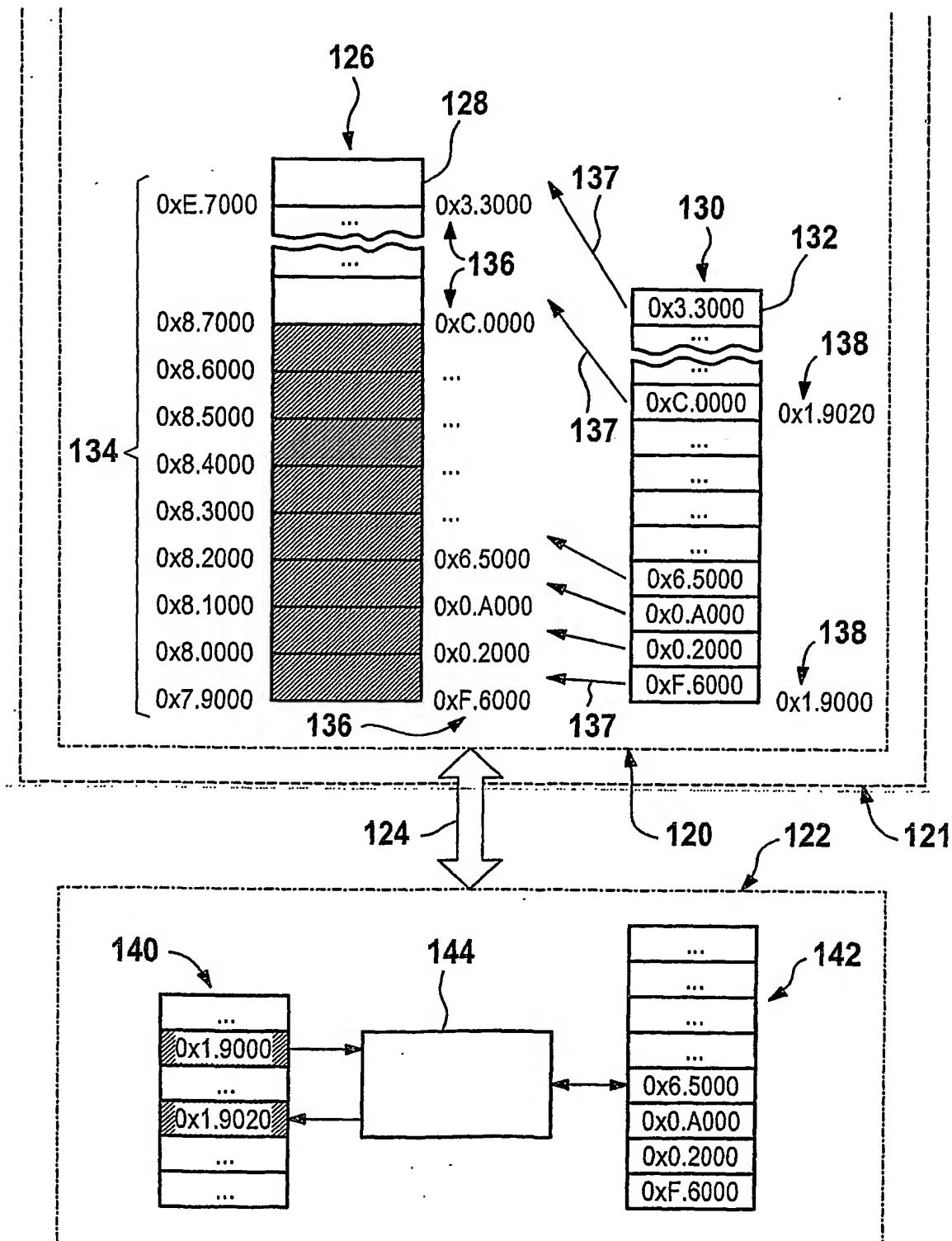
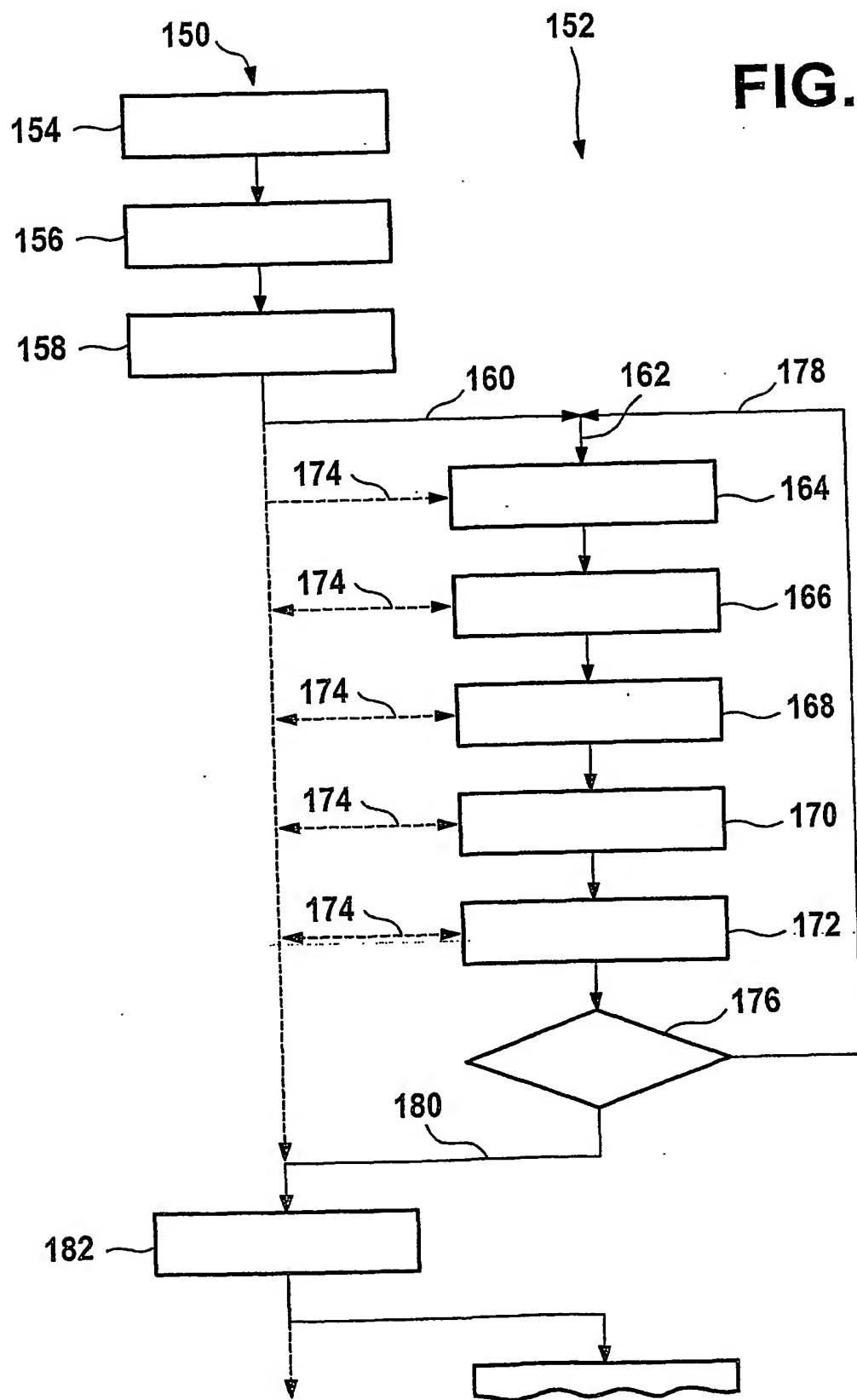


FIG. 5



4 / 4



INTERNATIONAL SEARCH REPORT

International Application No

PCT/EP 03/00138

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F13/32

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the International search (name of data base and, where practical, search terms used)

EPO-Internal, PAJ, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A X	US 5 890 220 A (ISHIKAWA TAKANORI ET AL) 30 March 1999 (1999-03-30) column 2, line 21 - line 29 column 2, line 50 - line 55 column 4, line 45 - line 47 ---	1-7, 14, 15 8-13
A	US 5 758 182 A (ROSENTHAL DAVID S H ET AL) 26 May 1998 (1998-05-26) abstract column 4, line 64 -column 5, line 4 column 6, line 52 - line 53 column 7, line 3 - line 8 column 8, line 13 - line 18 column 10, line 23 - line 28 column 10, line 34 - line 35 column 11, line 37 - line 61 ---	1-15

 Further documents are listed in the continuation of box C. Patent family members are listed in annex.

* Special categories of cited documents :

- *A* document defining the general state of the art which is not considered to be of particular relevance
- *E* earlier document but published on or after the International filing date
- *L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- *O* document referring to an oral disclosure, use, exhibition or other means
- *P* document published prior to the International filing date but later than the priority date claimed

- *T* later document published after the International filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

- *X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

- *Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

- *&* document member of the same patent family

Date of the actual completion of the International search

Date of mailing of the International search report

5 May 2003

14/05/2003

Name and mailing address of the ISA
European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl
Fax: (+31-70) 340-3016

Authorized officer

Albert, J

INTERNATIONAL SEARCH REPORT

International Application No

PCT/DE 03/00138

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2001/029556 A1 (IWAMOTO RICK ET AL) 11 October 2001 (2001-10-11) paragraph '0033! -----	1-15

INTERNATIONAL SEARCH REPORT
Information on patent family membersInternational Application No
PCT/GB 03/00138

Patent document cited in search report	Publication date	Patent family member(s)		Publication date
US 5890220	A	30-03-1999	JP	3124778 B2
			JP	4348442 A
			JP	4291438 A
US 5758182	A	26-05-1998	NONE	
US 2001029556	A1	11-10-2001	US	6282587 B1
			US	6092124 A

INTERNATIONALE RECHERCHENBERICHT

Internationales Aktenzeichen

PCT/DE 03/00138

A. KLASIFIZIERUNG DES ANMELDUNGSGEGENSTANDES
IPK 7 G06F13/32

Nach der Internationalen Patentklassifikation (IPK) oder nach der nationalen Klassifikation und der IPK

B. RECHERCHIERTE GEBIETE

Recherchierte Mindestprüfstoff (Klassifikationssystem und Klassifikationssymbole)

IPK 7 G06F

Recherchierte aber nicht zum Mindestprüfstoff gehörende Veröffentlichungen, soweit diese unter die recherchierten Gebiete fallen

Während der internationalen Recherche konsultierte elektronische Datenbank (Name der Datenbank und evtl. verwendete Suchbegriffe)

EPO-Internal, PAJ, WPI Data

C. ALS WESENTLICH ANGESEHENE UNTERLAGEN

Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
A X	US 5 890 220 A (ISHIKAWA TAKANORI ET AL) 30. März 1999 (1999-03-30) Spalte 2, Zeile 21 – Zeile 29 Spalte 2, Zeile 50 – Zeile 55 Spalte 4, Zeile 45 – Zeile 47 US 5 758 182 A (ROSENTHAL DAVID S H ET AL) 26. Mai 1998 (1998-05-26) Zusammenfassung Spalte 4, Zeile 64 – Spalte 5, Zeile 4 Spalte 6, Zeile 52 – Zeile 53 Spalte 7, Zeile 3 – Zeile 8 Spalte 8, Zeile 13 – Zeile 18 Spalte 10, Zeile 23 – Zeile 28 Spalte 10, Zeile 34 – Zeile 35 Spalte 11, Zeile 37 – Zeile 61 -- --	1-7, 14, 15 8-13 1-15
A		

Weitere Veröffentlichungen sind der Fortsetzung von Feld C zu entnehmen

Siehe Anhang Patentfamilie

- * Besondere Kategorien von angegebenen Veröffentlichungen :
- *A* Veröffentlichung, die den allgemeinen Stand der Technik definiert, aber nicht als besonders bedeutsam anzusehen ist
- *E* älteres Dokument, das jedoch erst am oder nach dem Internationalen Anmeldedatum veröffentlicht worden ist
- *L* Veröffentlichung, die geeignet ist, einen Prioritätsanspruch zweifelhaft erscheinen zu lassen, oder durch die das Veröffentlichungsdatum einer anderen im Recherchenbericht genannten Veröffentlichung belegt werden soll oder die aus einem anderen besonderen Grund angegeben ist (wie ausgeführt)
- *O* Veröffentlichung, die sich auf eine mündliche Offenbarung, eine Benutzung, eine Ausstellung oder andere Maßnahmen bezieht
- *P* Veröffentlichung, die vor dem Internationalen Anmeldedatum, aber nach dem beanspruchten Prioritätsdatum veröffentlicht worden ist

- *T* Spätere Veröffentlichung, die nach dem Internationalen Anmeldedatum oder dem Prioritätsdatum veröffentlicht worden ist und mit der Anmeldung nicht kollidiert, sondern nur zum Verständnis des der Erfindung zugrundeliegenden Prinzips oder der ihr zugrundeliegenden Theorie angegeben ist
- *X* Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann allein aufgrund dieser Veröffentlichung nicht als neu oder auf erforderlicher Tätigkeit beruhend betrachtet werden
- *Y* Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann nicht als auf erforderlicher Tätigkeit beruhend betrachtet werden, wenn die Veröffentlichung mit einer oder mehreren anderen Veröffentlichungen dieser Kategorie in Verbindung gebracht wird und diese Verbindung für einen Fachmann naheliegend ist
- *&* Veröffentlichung, die Mitglied derselben Patentfamilie ist

Datum des Abschlusses der Internationalen Recherche	Absendedatum des Internationalen Recherchenberichts
5. Mai 2003	14/05/2003
Name und Postanschrift der Internationalen Recherchenbehörde Europäisches Patentamt, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax (+31-70) 340-3016	Bevollmächtigter Bediensteter Albert, J

INTERNATIONALER ~~SEARCH~~ **RECHENBERICHT**

Internationales Aktenzeichen

PCT/DE 03/00138

C.(Fortsetzung) ALS WESENTLICH ANGESEHENE UNTERLAGEN

Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
A	US 2001/029556 A1 (IWAMOTO RICK ET AL) 11. Oktober 2001 (2001-10-11) Absatz '0033! -----	1-15

BEST AVAILABLE COPY

INTERNATIONALER RECHERCHENBERICHT

Angaben zu Veröffentlichungen, die zu der in Patentfamilie gehören

Internationales Aktenzeichen

PCT/DE 03/00138

Im Recherchenbericht angeführtes Patentdokument	Datum der Veröffentlichung	Mitglied(er) der Patentfamilie		Datum der Veröffentlichung
US 5890220	A 30-03-1999	JP	3124778 B2	15-01-2001
		JP	4348442 A	03-12-1992
		JP	4291438 A	15-10-1992
US 5758182	A 26-05-1998	KEINE		
US 2001029556	A1 11-10-2001	US	6282587 B1	28-08-2001
		US	6092124 A	18-07-2000

BEST AVAILABLE COPY

(12) NACH DEM VERTRAG ÜBER DIE INTERNATIONALE ZUSAMMENARBEIT AUF DEM GEBIET DES
PATENTWESENS (PCT) VERÖFFENTLICHTE INTERNATIONALE ANMELDUNG

BERICHTIGTE FASSUNG

(19) Weltorganisation für geistiges Eigentum
Internationales Büro



(43) Internationales Veröffentlichungsdatum
4. September 2003 (04.09.2003)

PCT

(10) Internationale Veröffentlichungsnummer
WO 2003/073301 A1

(51) Internationale Patentklassifikation⁷: **G06F 13/32**

(21) Internationales Aktenzeichen: PCT/DE2003/000138

(22) Internationales Anmeldedatum:
20. Januar 2003 (20.01.2003)

(25) Einreichungssprache: Deutsch

(26) Veröffentlichungssprache: Deutsch

(30) Angaben zur Priorität:
102 08 300.2 26. Februar 2002 (26.02.2002) DE

(71) Anmelder (für alle Bestimmungsstaaten mit Ausnahme von US): ROBERT BOSCH GMBH [DE/DE]; Postfach 30 02 20, 70442 Stuttgart (DE).

(72) Erfinder; und

(75) Erfinder/Anmelder (nur für US): BARTHEL, Joerg [DE/DE]; An der Innerste Au 22, 31139 Hildesheim (DE). BEHME, Dirk [DE/DE]; Hachmeisterstrasse 24, 31139 Hildesheim (DE).

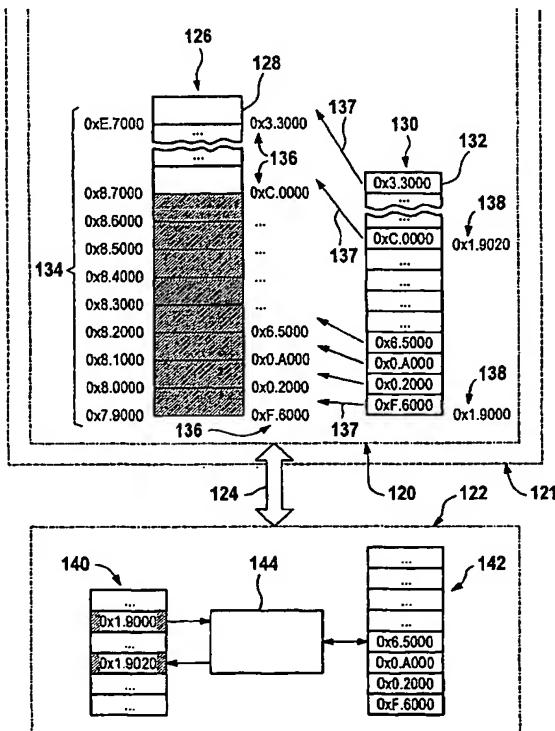
(81) Bestimmungsstaaten (national): AT, AU, BR, CA, CH, CN, CZ, ES, FI, GB, IN, JP, KR, LU, MX, PL, RU, SE, SK, US, ZA.

(84) Bestimmungsstaaten (regional): europäisches Patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, SE, SI, SK, TR).

[Fortsetzung auf der nächsten Seite]

(54) Title: METHOD FOR TRANSFERRING DATA VIA A DATA BUS

(54) Bezeichnung: VERFAHREN ZUM ÜBERTRAGEN VON DATEN ÜBER EINEN DATENBUS



(57) Abstract: The invention relates to a method for transferring data via a data bus (124), in particular via a PCI bus (124). According to the method, a second additional memory area (130) is created in a main memory (120) of a processor, in which all physical addresses (136, 138) of the memory pages (128) are filed. During the data transfer, the required physical addresses are transferred independently.

(57) Zusammenfassung: Es ist ein Verfahren zum Übertragen von Daten über einen Datenbus (124), insbesondere über den PCI-Bus (124), beschrieben. Das erfindungsgemäße Verfahren sieht vor, dass in einem Hauptspeicher (120) einer Recheneinheit ein zweiter zusätzlicher Speicherbereich (130) angelegt wird, in den alle physikalischen Adressen (136, 138) der Speicherseiten (128) abgelegt werden. Während der Datenübertragung werden die benötigten physikalischen Adressen selbstständig übertragen.

WO 2003/073301 A1

BEST AVAILABLE COPY



Veröffentlicht:

— *mit internationalem Recherchenbericht*

**(48) Datum der Veröffentlichung dieser berichtigten
Fassung:**

5. August 2004

Zur Erklärung der Zweibuchstaben-Codes und der anderen Abkürzungen wird auf die Erklärungen ("Guidance Notes on Codes and Abbreviations") am Anfang jeder regulären Ausgabe der PCT-Gazette verwiesen.

(15) Informationen zur Berichtigung:

siehe PCT Gazette Nr. 32/2004 vom 5. August 2004, Section II

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- BLACK BORDERS**
- IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- FADED TEXT OR DRAWING**
- BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- SKEWED/SLANTED IMAGES**
- COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- GRAY SCALE DOCUMENTS**
- LINES OR MARKS ON ORIGINAL DOCUMENT**
- REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.